

Online Multicamera Tracking with a Switching State-Space Model

Wojciech Zajdel A. Taylan Cemgil Ben J.A. Kröse
University of Amsterdam, Intelligent Autonomous Systems
1098SJ Amsterdam , Kruislaan 413, The Netherlands
{wzajdel, cemgil, krose}@science.uva.nl

Abstract

The paper presents a novel method for online tracking of multiple objects with non-overlapping cameras. The method is based on a generative model defining probabilistic dependencies between observations, the underlying color properties of objects and their dynamics. It allows for a full Bayesian inference of trajectories. We developed an on-line algorithm for efficient, approximate inference and we demonstrate it to be accurate in an office environment.

1. Introduction

Object tracking in wide areas often relies on a network of cameras that have disjoint fields of view (FOVs). In this setup, every camera provides only a local description of objects appearing within its FOV. Global trajectories of objects are recovered by association of their local appearances at various FOVs. This is a hard association problem, since objects may appear at varying viewing angles and under different illumination. Moreover, the motion of objects between distant FOVs is irregular (non-smooth).

Existing techniques search for an optimal solution by explicit considering of likelihood of alternative trajectories. Unless the application domain constraints possible associations [7], the space of trajectories is intractable [1]. Therefore, the association is usually solved approximately, using Markov chain Monte Carlo (MCMC) [9] sampling or multiple hypothesis trackers (MHT) [3]. Sampling methods are well suited to approximate some traffic statistics, such as travel time between FOVs, however, they are not designed for online applications. MHT is a deterministic, online method that prunes trajectories with low likelihood conditioned on the incoming data. However, pruning does not preserve any information corresponding to the discarded trajectories. In noisy environments, such information could be necessary for proper association of the future data.

The idea underlying our approach is to identify each object with an unique label, and treat the sequence of local

appearances as noisy observations from the hidden labels. The probabilistic dependency of labels, appearances and motion constraints is formulated as a dynamical generative model. Conditioned on the observations, we compute posterior probabilities of the labels using the Bayes rule. However, due to the inherent association ambiguity the posterior densities in our model take the form of mixtures with an intractable number of components. Our approximation does not discard components corresponding to unlikely associations, but replaces the mixture with a tractable family in a way that preserves the moments of the exact posterior density. The resulting online algorithm falls into the class of deterministic assumed-density filtering approximations [2].

2. Generative Model

In this paper we focus on tracking people in a building, and assume that data from all cameras are processed centrally. Our generative model defines probability density of local appearances (called *observations*) conditioned on the persons' unique labels. Such a conditional density is used to find posterior distributions on the labels given the observations. The trajectories are recovered by taking the most likely label for every observation.

Let $Y_k = \{O_k, D_k\}$ denote the k th (in time order) observation from any camera, where O_k describes d -dimensional color features computed while a person was visible within FOV; and D_k include the discrete camera location, the time tag and the borders of entering and leaving FOV (left, right).

Although the underlying color properties of a person do not change, the features O differ whenever the person is observed due to the varying pose or illumination. To model the effects of such variations we assume that each person is a color process and O is a sample from a Normal pdf specific to the person. For every observed O_k we introduce a latent variable $X_k = \{\mathbf{m}_k, \mathbf{V}_k\}$ that represents parameters of Normal density (kernel) from which O_k is sampled. The $d \times 1$ vector \mathbf{m} describes the person's specific, expected features. The $d \times d$ covariance matrix \mathbf{V} tells how sensitive the person's appearance is to the changing observation condi-

tions. For instance, the appearance a person dressed uniformly in black is relatively independent of illumination or pose, so his/her covariance has small eigenvalues. The appearance of a person dressed in white or non-uniform colors is easily affected by pose or illumination, so we model such a person with a 'broad' kernel.

We set a prior density $\pi(X)$ for the latent parameters $X = \{\mathbf{m}, \mathbf{V}\}$. A convenient joint model for the mean and covariance is the 'Normal-Inverse Wishart' [6] pdf

$$\pi(X) = \phi(X|\theta_0) = \mathcal{N}(\mathbf{m}; \mathbf{a}_0, \kappa_0 \mathbf{V}) \mathcal{IW}(\mathbf{V}; \eta_0, \mathbf{C}_0) \quad (1)$$

where $\theta_0 = \{\mathbf{a}_0, \kappa_0, \eta_0, \mathbf{C}_0\}$ are hyperparameters.

We refer to the component D (location, time, borders) as *dynamics* and assume that it is observed noise-free. A sequence $\{D_1^{(n)}, D_2^{(n)}, \dots\}$ of such data assigned to n th person (superscript) defines a path in the building. We model the path, i.e., the sequence $\{D_1^{(n)}, D_2^{(n)}, \dots\}$, as a random, 1st-order Markov process. The path is started by sampling from an initial distribution P_{δ_0} , and extended by sampling from $P_{\delta}(D_{i+1}^{(n)}|D_i^{(n)})$. The distributions P_{δ} and P_{δ_0} follow from the topology of FOVs. For instance, P_{δ} could indicate the likelihood of moving from a FOV to another FOV, and P_{δ_0} could be the likelihood of starting a path at some FOV.

Association variables For every observation Y_k there is a corresponding variable S_k that denotes the label of the person to which Y_k is assigned. Within the first k data, $Y_{1:k} \equiv \{Y_1, \dots, Y_k\}$, there may be at most k different people, so S_k has k different states; $S_k \in \{1, \dots, k\}$. The label S_k is accompanied by auxiliary variables: a counter C_k , and pointers $Z_k^{(1)}, \dots, Z_k^{(k)}$. The counter, $C_k \in \{1, \dots, k\}$, indicates the number of trajectories present in the data $Y_{1:k}$. The n th pointer variable, $Z_k^{(n)} \in \{0, \dots, k-1\}$, denotes the time index when the n th person was last observed *before* time-slice k . Value $Z_k^{(n)} = 0$ indicates that the person n has not yet been observed. At the k th time-slice, there can be up to k persons, so we need $Z_k^{(n)}$ for $n = 1, \dots, k$. Note, that the auxiliary variables provide immediate 'lookup' reference to the information that is already encoded by $S_{1:k}$.

We initialize the counter $C_0 = 0$, and use the association variables to describe the dependency of observations on the labels. To generate Y_k , $k \geq 1$, we select a label S_k . People enter FOVs irregularly, so we choose uniformly between observing one of the known C_{k-1} persons or a new one;

$$S_k \sim \text{Uniform}(1, \dots, C_{k-1}, C_{k-1} + 1). \quad (2)$$

Next, we deterministically update the auxiliary variables:

$$C_k = C_{k-1} + [S_k > C_{k-1}], \quad (3)$$

$$Z_k^{(k)} = 0, \quad (4)$$

$$Z_k^{(n)} = Z_{k-1}^{(n)} [S_{k-1} \neq n] + (k-1) [S_{k-1} = n], \quad (5)$$

where $n = 1, \dots, k-1$. The symbol $[f]$ is a binary indicator; $[f] \equiv 1$ iff the proposition f is true, and $[f] \equiv 0$ otherwise. If we introduced a new person, $S_k = C_{k-1} + 1$, then the counter C_k has to be increased (3). The pointers summarize associations before k , so we update them using S_{k-1} . A person labeled as k cannot be observed before time k , so the pointer to his/her previous observation, $Z_k^{(k)}$ is set to zero. The pointer to the last observation of the n th, $n < k$, person either does not change or we set it to the index of the preceding observation, $k-1$, only if $S_{k-1} = n$.

Next, we generate the latent parameters X_k of the person indicated by S_k . If this person has been already observed, then the index of his/her last observation $Z_k^{(S_k)}$ is nonzero. By our assumption the latent parameters do not change, so we simply copy X from the previous instance. If the current person is observed for the first time, $Z_k^{(S_k)} = 0$, then we sample X from the prior $\pi(X)$;

$$X_k = X_{Z_k^{(S_k)}} [Z_k^{(S_k)} > 0] + X^{\text{new}} [Z_k^{(S_k)} = 0], \quad (6)$$

$$X^{\text{new}} \sim \pi. \quad (7)$$

Finally, we render the observation $Y_k = \{O_k, D_k\}$ given the parameters of a kernel $X_k = \{\mathbf{m}_k, \mathbf{V}_k\}$ and the pointer to the past dynamics of the current object, $Z_k^{(S_k)} = i$;

$$O_k \sim \mathcal{N}(\mathbf{m}_k, \mathbf{V}_k), \quad (8)$$

$$D_k \sim P_{\delta}(D_k|D_i) [i > 0] + P_{\delta_0}(D_k) [i = 0]. \quad (9)$$

Graphical model Figure 1 shows a graphical representation of our model. At the k th time-slice, a single variable H_k denotes all association variables; $H_k \equiv \{S_k, C_k, Z_k^{(1)}, \dots, Z_k^{(k)}\}$. This variable evolves as a Markov process with a transition distribution $p(H_k|H_{k-1})$ following from (2)–(5). The kernel parameters evolve as a mixed memory Markov model [8] with a transition distribution $p(X_k|X_{1:k-1}, H_k)$ following from (6)–(7). The past variables $X_{1:k-1}$ become a 'memory' and H_k a discrete 'switch' selecting one X from $X_{1:k-1}$. New parameters X_k are generated using the selected variable or sampled from the prior. For clarity, in Fig. 1 we skipped the edges representing dependencies of dynamics $D_{1:k}$.

3. Online tracking by filtering

When a person completes a pass through a FOV, an observation Y_k arrives. We use our model to estimate the label S_k by probabilistic filtering, i.e., computing a posterior density on the latent variables (including S_k) given the available data $Y_{1:k}$. From Fig. 1 we see, that in our model the influence of the past data $Y_{1:k-1}$ is mediated by the latent variables H_{k-1} and $X_{1:k-1}$, so we need a density on these variables conditioned on $Y_{1:k-1}$. We compute such a density after receiving every Y_k (when k increases), thus our *filtered*

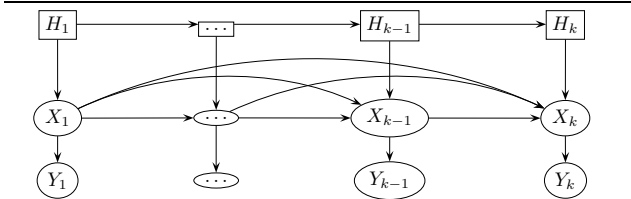


Figure 1. A graphical model.

density is $p(X_{1:k}, H_k | Y_{1:k})$. This density provides the necessary information to estimate the current label and to process future labels. First, we find a predictive density

$$p_r(X_{1:k}, H_k) = \sum_{H_{k-1}} p(H_k | H_{k-1}) p(X_k | X_{1:k-1}, H_k) \times p(X_{1:k-1}, H_{k-1} | Y_{1:k-1}), \quad (10)$$

where the last term comes from the previous iteration. The filtered density is found by updating $p_r(X_{1:k}, H_k)$ with Y_k :

$$p(X_{1:k}, H_k | Y_{1:k}) = \frac{1}{L_k} p(Y_k | X_k, H_k) p_r(X_{1:k}, H_k) \quad (11)$$

where L_k is a normalization term, and $p(Y_k | X_k, H_k)$ follows from (8)–(9). The dependency of Y_k on dynamics $D_{1:k-1}$ is not written explicitly, however it is always assumed.

Repetitive summation over labels in (10) yields a density on parameters X that is mixture of $\mathcal{O}(k!)$ component pdfs, because the continuous variables X depend on the combination of all labels $S_{1:k}$. Similar to the other switching state-space models, the exact computation is intractable. There are approximate inference methods applicable for such models (cf. [8]). We follow the assumed-density filtering (ADF) approach for it is suited for online implementations.

Representation ADF [2] approximates the filtered distribution with a tractable family. We choose a family

$$p(X_{1:k}, H_k | Y_{1:k}) \approx q(S_k, C_k) \prod_{i=1}^k q_k(X_i) q(Z_k^{(i)}) \quad (12)$$

that factorizes the discrete variables from the continuous. Approximating the joint distribution on H_k with a product of simpler models sidesteps maintaining a large table with probabilities for every combination of their states. Each continuous variable, X_i , represents parameters of a Gaussian kernel. We approximate the posterior on this variable with ‘Normal-Inverse Wishart’ family; $q_k(X_i) = \phi(X_i | \theta_{i,k})$, where $\theta_{i,k}$ are hyperparameters specific to the i th kernel after k filtering steps. (This family is conjugate to the Normal density [6].)

One-step ADF update When Y_k arrives we only have an approximation $\tilde{p}(X_{1:k-1}, H_{k-1} | Y_{1:k-1})$ of the filtered density in the form of (12). Executing (10)–(11) yields $p(X_{1:k}, H_k | Y_{1:k})$ that is not in the assumed family. ADF projects it to such member of the family that offers the closest approximation in the Kullback-Leibler (KL) sense. The nearest in the KL-sense factorial distribution is the product of marginals [4], so we recover the representation (12) by computing the marginals of $p(X_{1:k}, H_k | Y_{1:k})$ from (11).

The detailed marginals are provided in [10]. We note that the marginalization is efficient for two reasons. First, our model is sparse. We see from (6)–(9): when we set a label $S_k = m$ then out of all auxiliary variables, only $Z_k^{(m)}$ is referenced. If we set $Z_k^{(m)} = j$, only parameters X_j and dynamics D_j contribute to Y_k . Thus, when we marginalize (11), the variables that do not affect Y_k integrate out from the predictive density $p_r(H_k, X_{1:k})$. We do not find the joint prediction $p_r(H_k, X_{1:k})$, only its necessary marginals. Secondly, finding the marginals of the predictive distribution is simple, because the variables $H_k, X_{1:k}$ evolve deterministically (except for S_k , and X_k when it is sampled from prior). A marginal $q_k(X_i)$ is a mixture of k pdfs, each corresponding to a different label S_k . We approximate the mixture with a single, KL-nearest density from the assumed family.

4. Experiments

We test our method using 70 observations (local appearances) of 5 persons, who were observed with 7 non-overlapping cameras in an office-like environment. In the first experiment we measure tracking accuracy of our approach and compare it with the MCMC and MHT methods. In the second experiment we compare our method with a simple approach that clusters observations on the basis of appearance proximity.

Color features O From a video sequence with a pass of a person through a FOV we selected a single frame. In this frame we manually found the pixels representing the person and transformed the original RGB pixels into a color-channel normalized space [5] to suppress the effects introduced by the color of illuminating light. The person’s image was split into three fixed regions as in Fig. 2a). The regions are a heuristic for describing people. For each region we computed the 3D average color, obtaining in total a 9D feature. Unlike a color histogram, such features provide a low dimensional summary of color content together with its geometrical layout.

We set the parameters $\theta_0 = \{\mathbf{a}_0, \kappa_0, \eta_0, \mathbf{C}_0\}$ of the prior (1) $\pi(X)$ as follows: the expected features $\mathbf{a}_0 = \mathbf{0}$ (9D zero vector); the scale $\kappa_0 = 100$; the degrees of freedom $\eta_0 = 9$ (dimension of features), $\mathbf{C}_0 = 10^{-3}\mathbf{I}$, where

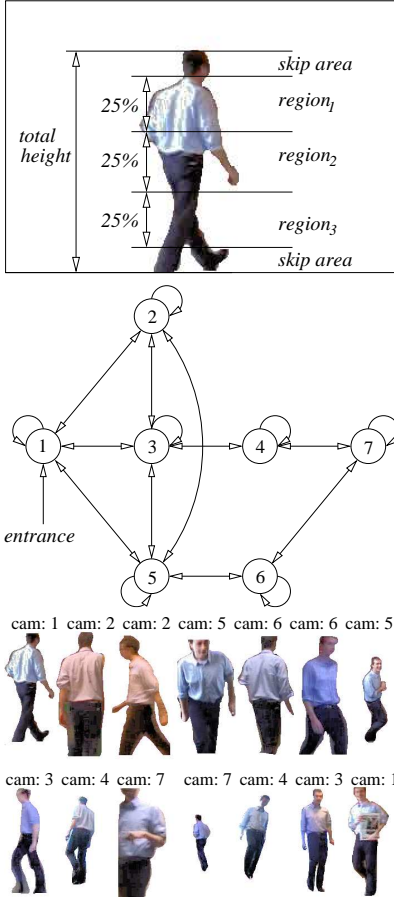


Figure 2. top) Regions for computing color features. center) The movement constraints between 7 cameras represented by P_δ and P_{δ_0} ; arrows indicate possible paths. bottom) An example of a recovered trajectory; images (top left to bottom right) indicate the sequence of appearances at FOVs.

\mathbf{I} is a 9D identity matrix. Matrix \mathbf{C}_0 with small eigenvalues indicates that we expect relatively 'sharp' kernels. Since the means \mathbf{m} are not known, we set the scale κ_0 to a large value.

Dynamic features A pass of a person through a FOV is described with dynamic features $D = \{L, E, Q, T\}$. The term L is the camera location; $L \in \{1, 2, 3, 4, 5, 6, 7\}$; T is the time of observation; variable E (Q) denotes the frame border through which the object enters (leaves) the camera FOV. Variables E, Q took the values: 'left', 'right', 'other'.

Our model P_δ of Markovian paths (transitions) is $p(L_n, E_n | L_p, Q_p) p(T_n | T_p)$. The first quantity is the probability of moving directly from location L_p when quitting the FOV via border Q_p to location L_n and enter-

| | Our method | MHT | MCMC 10 ³ samples | MCMC 10 ⁴ samples |
|---------|------------|-----|---------------------------------|---------------------------------|
| error | 5% | 21% | 20% ± 0.6 | 12% ± 0.5 |
| objects | 5 | 8 | 6.5 ± 0.3 | 6.5 ± 0.3 |

Table 1. Average classification errors and estimated number of objects (trajectories) for the MCMC, MHT and our method.

ing the FOV via border E_n . The possible transitions in our environment are sketched in Fig. 2b). All transitions marked with an arrow have equal non-zero probability, all other have zero probability. The term $p(T_n | T_p)$ is the probability of being observed at time T_n if the previous observation was at time T_p regardless of the locations. We used a binary model $p(T_n | T_p)$ that prevents a person appearing at different locations at the same time. The distribution $P_{\delta_0}(L, E)$ gives the likelihood of starting a path at FOV L via border E . In our case it was only possible via left border of FOV 1.

4.1. Experiment 1

From the videos we manually recovered the true trajectories, which we marked with *true* labels. We evaluated our method by comparing the estimated trajectories with the true ones. The tracking accuracy is defined as follows. Given an *estimated* trajectory we count the *true* labels of observations, and assume that this trajectory describes a person with the most frequent label. The observations with other labels within this trajectory are considered as wrongly classified. The ratio of wrongly classified to all observations within the trajectory makes the classification error. Tracking accuracy is measured by the average of such errors over all estimated trajectories. An additional criterion is the number of estimated trajectories.

The MHT and MCMC algorithms used the same models for appearance and dynamics as our method. The MCMC method [9] samples partitions of the data set, where a single partition defines trajectories of all hypothetical objects. After sampling 10³ and 10⁴ partitions we took the partition with the highest posterior probability as a solution.

The tracking accuracy of the compared methods is summarized in Table 1. The results for the MCMC method are shown as means from ten runs. We observe that our algorithm found the correct number of persons and returned nearly exact trajectories. An example of the estimated trajectory by our method is given in Fig. 2c). The accuracy of the trajectories estimated with MCMC or MHT method does not match with the accuracy of trajectories found by our method. Moreover, MCMC and MHT overestimated the number of distinct persons(trajectories) in the data set.

4.2. Experiment 2

A simple method for tracking is to ignore the dynamics and cluster the observations on the basis of appearance similarities. Below we compare the clusters obtained in the space of color features in two cases; (i) when the association was based only on the color features, and (ii) when the association was based on dynamics and color features.

After processing observations our algorithm maintains parameters $X = \{\mathbf{m}, \mathbf{V}\}$ of Gaussian kernels representing the latent color properties. The j th kernel is represented by a distribution $\phi(X_j|\theta_{j,70})$ conditioned on 70 observations, because filtering updates distributions on all parameters in the memory. Given the hyperparameters $\theta_{j,70}$ we find the expected mean $E[\mathbf{m}_j]$ and the expected covariance $E[\mathbf{V}_j]$ of the j th kernel; $E[\mathbf{m}_j] = \mathbf{a}_{j,70}$ and $E[\mathbf{V}_j] = \mathbf{C}_{j,70}/(\eta_{j,70} - d - 1)$. The kernels and the features have $d = 9$ dimensions. For visualization, we find a 2 dimensional PCA projection of the original data. Figure 3 presents the projections of the observed features (as points) and the expected kernels (as ellipses). Figure 3b) shows that when association is based only on appearance, then the kernels form only two clusters. In Fig. 3a) we see that when the dynamic features supported tracking, then the estimated kernels closely correspond to the latent appearance features of the persons.

5. Conclusions

We described a technique for tracking multiple objects in wide areas with non-overlapping cameras. It is a deterministic alternative to the tracking methods that use stochastic sampling. Our method applies an approximate Bayesian inference algorithm for estimating the solution to the intractable data association problem. We explicitly model and estimate the the number of objects on the basis of movement constraints and appearance proximity. The described appearance noise models and the approximate association algorithm are particularly meant for difficult environments, where the tracked objects appear irregularly under non-uniform illumination or pose. In such an environment, the proposed method performed superior to the standard multiple hypotheses tracker or sampling methods.

Acknowledgments This research is supported by the Technology Foundation STW (grant no ANN.5312), applied science division of NWO and the technology program of the Dutch Ministry of Economic Affairs.

References

[1] Y. Bar-Shalom and X.-R. Li. *Estimation and Tracking: Principles, Techniques and Software*. Artech House, 1993.

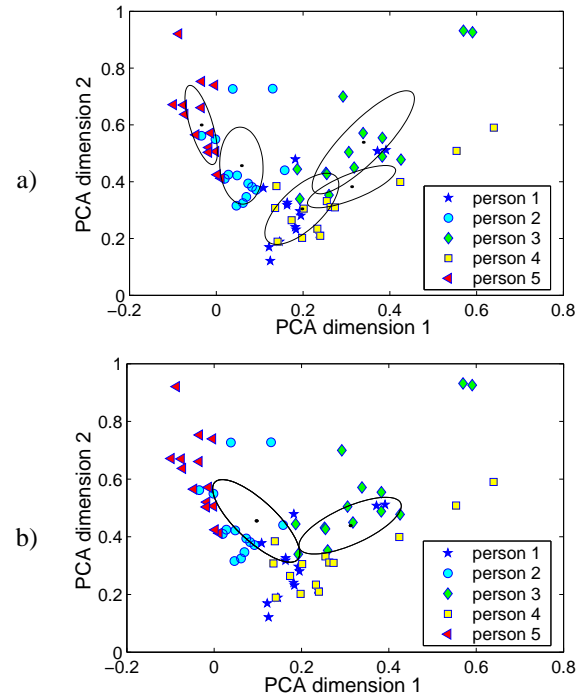


Figure 3. The kernels obtained when tracking was based: a) on dynamics and appearance; b) only on appearance.

- [2] X. Boyen and D. Koller. Tractable inference for complex stochastic processes. In *Proc. of Conf. on Uncertainty in Artificial Intelligence*, pages 33–42. Morgan Kaufman, 1998.
- [3] R. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade. Algorithms for cooperative multisensor surveillance. *Proceedings of the IEEE*, 89(10):1456–1477, October 2001.
- [4] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, 1991.
- [5] M. Drew, J. Wei, and Z. Li. Illumination-invariant color object recognition via compressed chromaticity histograms of color-channel-normalized images. In *Proc. of Int. Conf. on Computer Vision*, pages 533–540, 1998.
- [6] A. Gelman, J. B. Carlin, H. S. Stern, and D. D. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 1995.
- [7] T. Huang and S. Russell. Object identification: A Bayesian analysis with application to traffic surveillance. *Artificial Intelligence*, 103(1–2):1–17, 1998.
- [8] K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, UC, Berkeley, 2002.
- [9] H. Pasula, S. Russell, M. Ostland, and Y. Ritov. Tracking many objects with many sensors. In *Proc. of Int. Joint Conf. on Artificial Intelligence*, pages 1160–1171, 1999. Stockholm.
- [10] W. Zajdel, A. Cemgil, and B. Kröse. Technical report: A hybrid graphical model for online multicamera tracking. Technical report, University of Amsterdam, 2003.